

A uniform intermediate model for high-fidelity and efficient visualization of multidisciplinary heterogeneous simulation data

Yanli Shao^a, Yusheng Liu^{a*}, Xiaoping Ye^b, Hongwei Wang^c

^a*State Key Lab. of CAD&CG, Zhejiang University, Hangzhou, P.R.China, 310027*

^b*Lishui University, Lishui, P.R. China, 323000*

^c*School of Engineering, University of Portsmouth, Portsmouth, UK, PO1 3DJ*

(Email: ysliu@cad.zju.edu.cn)

Simulation-driven design has become an effective methodology for improving the design process, in terms of both time and cost. Visualization of simulation results with high efficiency and fidelity is of significant importance for this methodology especially for the multidisciplinary simulation of complex products. One of the key challenges is to display and view multidisciplinary simulation results that are heterogeneous and tool-specific. In this research, a uniform intermediate model which supports high-fidelity and efficient visualization of multidisciplinary heterogeneous simulation data is proposed and developed. Firstly, the problem of generating a uniform intermediate mesh model is formalized and represented as an optimization model. Then, the Genetic Algorithm (GA) is employed and improved to optimize the uniform intermediate mesh model globally and locally after the direct initial (D-initial) mesh model is refined based on the hybrid mesh size field. Finally, the mapping and interpolation relationships between the uniform intermediate mesh model and the mesh models from different disciplines are established to implement simulation results visualization in a CAD platform. To improve computational efficiency, two strategies are deliberately specifically employed, namely indexing and parallel computing. Several experiments are conducted to demonstrate the performance of the proposed algorithm.

Keywords: Intermediate mesh model; Mesh optimization; Genetic algorithm (GA); Hybrid mesh size field

1. Introduction

Computer-Aided Design (CAD) and Computer-Aided Engineering (CAE) are imperative for the development of complex products. Simulation, as an effective alternative of physical testing in terms of both cost and time, allows more design options to be considered in a shorter time (Sellgren 1994, 1995). With the continuous improvement of computational capability, simulation is increasingly used for the development of complex products to reduce the development costs, shorten the time to market and improve the product quality, all of which are the key factors for achieving competitive advantages (Nolan et al. 2013). Therefore, simulation-driven design is becoming a key methodology for product design and development.

The simulation tasks for complex product development generally involve multiple domains such as structural field analysis, flow field analysis, thermal field analysis etc. Thus, various simulation

systems, including ANSYS, Abaqus, MSC Nastran etc., have been developed to the needs of developing domain-specific simulations (Cote et al. 2004). However, one of the key challenges for implementing a simulation-driven design methodology is the absence of an engineering environment and a standard process that are readily available for the design team. Firstly, the simulation results obtained using different packages are heterogeneous and tool-specific. Moreover, product design and simulation development are generally carried out in different departments in many companies and as such simulation systems may not be available in the design department due to cost consideration. Hence, it is difficult for designers to conveniently view the visualized simulation results even though simulation data can be sent to them quickly (Shao et al. 2013). Secondly, the simulation data for different disciplines sometimes need to be visualized on the same model at the same time to show coupling of different fields (Liu et al. 2010). These

deficiencies have resulted in an apparent failure of improving design problem-solving by efficiently and effectively utilizing simulation data. Therefore, this necessitates research work on high-fidelity heterogeneous simulation data visualization based on a uniform model that enables simulation data for different disciplines to be maintained independently and shared effectively.

This research is precisely motivated by this gap and aims to develop and evaluate such a uniform intermediate mesh model. It has a particular focus on novel methods on the optimization of this intermediate model as well as on the mapping and construction of geometric data. The rest of the paper is organized as follows. Section 2 gives a review of related work. Section 3 discusses the motivation and methodology of this work. Section 4 details the formulation of the optimization problem for the uniform intermediate mesh model. The GA-based optimization of the uniform intermediate mesh model and the generation of a Refined initial (R-initial) intermediate mesh model are described in Section 5 and Section 6, respectively. The mapping and interpolation between the data in the uniform intermediate mesh model and the data in discipline-specific mesh models are described in Section 7. Section 8 discusses implementation and evaluation of the model based on simulation experiments. Finally, Section 9 outlines the main conclusions of this work and discusses future work.

2. Related Work

The effective integration and visualization of multidisciplinary simulation data is crucial for developing a collaborative design environment for complex product development. Some research has been done to facilitate high-fidelity visualization of multidisciplinary simulation results so as to enable designers to conveniently and efficiently view simulation data. For example, Immersive SIM Engineering (2008) developed a CAE visualization tool called Immersive SIGHT for the visualization and sharing of CAE data. Specifically, it translates CAE analysis results into a structure based on the Virtual Reality Modeling Language (VRML) or X3D. This

tool is limited to only three domains and supports only a few functions pertaining to the translation and visualization of CAE data. Park and Kim (2012) presented a sharable and neutral format, namely practical post-analysis (PAM), which allowed the efficient sharing and visualization of finite element analysis (FEA) data in a collaborative design process through hierarchical data management using a single structure. It achieves substantial data compression by storing only essential finite element (FE) information as well as efficient visualization of heterogeneous analytic results by using a modified scene graph data structure. However, this work is mainly focused on the efficient exchange of FEA data produced in variety of CAE tools and cannot achieve high-fidelity visualization of integrated multidisciplinary simulation data. Liu, Fu, and Tan. (2010) presented a novel method of integration and visualization for FEA data of multiple physical fields. This method emphasizes the coupling of geometrical and physical information from heterogeneous mesh models and the interpolation of simulation results. Even though the integrated visualization of multidisciplinary simulation data can be achieved based on a transitional mesh, it is relatively complicated to construct the transitional mesh and reconstruct geometric models.

Isogeometric analysis (IGA) (Hughes et al. 2005; Bazilevs et al. 2006; Cottrell et al. 2009) is a generalization of classical FEA and proves to possess improved properties. Basic functions generated from non-uniform rational B-splines (NURBS) are employed to construct a precise geometric model. This eliminates geometrical errors and the necessity of linking the refinement procedure to a CAD representation of the geometry, as in classical FEA. However, even though the geometric representation of design and analysis models are exactly the same in IGA, it is still necessary to construct a uniform model to realize the integrated visualization of multidisciplinary simulation results due to the fact that the geometric models vary in different simulation requirements. Zhang et al. (2014) developed a Rotor Multidisciplinary Simulation Integration (RotorMSI) platform to solve the problem of multidisciplinary

process integration and resource-sharing of the rotor blade design process. A multidisciplinary simulation integration framework and a flow model are proposed to establish the integrated RotorMSI platform. The framework model of two-level flows and a group of nodes are presented to integrate tools for different disciplines by analyzing the numerical process of the rotor blade design. This platform is only applicable to the helicopter design industry due to the fact that the method developed is particularly focused on motor design.. In addition, it is hard to view integrated multidisciplinary simulation data as simulation results of different disciplines are visualized respectively.

Another challenge is to exchange detailed simulation information for checking and analysis so as to allow designers to make correct design changes conveniently and accurately. Hence, many previous research efforts have been focused on data exchange and integration. The standard for exchange for product model data (STEP) (Kemmerer 1999) AP209 (ISO 2001) has been proposed to enable CAE data exchange for various systems, which specifies formats for the data associated with simulation software and is suitable for dealing with analytical information (Hunten 2000). Many CAE studies have examined the feasibility of analysis data exchange using AP209 (Hunten 1997; Charles et al. 2005; MSC 2011). However, as a standard protocol, it accommodates all the data from both CAD and CAE, which easily increases the sizes of data files and thus make it difficult to be exchanged. As a result, research work has also been done on lightening CAE data to minimize the file size. Song et al. (2009) developed a CAE data exchange method, which relied on heterogeneous CAE systems, a virtual reality system, and a lightweight CAE middleware called CAE2VR for the effective sharing of geometric and analysis data. The generic CAE kernel of the CAE2VR middleware provides a function library including the creation of geometric elements and the evaluation and visualization of analysis data. However, the CAE data can be visualized only in a VR system or in a VRML Web browser. Cho et al. (2011) proposed a data representation scheme which covered most of the typical CAE data in a unified manner to reduce the size

of large simulation data in order to enhance the management and sharing of simulation information in collaborative environments. They incorporated FEA data into their previous research on BST, developed an integrated engineering data representation, and suggested an extended BST. Since reduction of data size happens when data is converted again to a BST data, the resulting FEA data become more convenient to store and share for engineers working in an integrated manner. However, further studies on more diverse simulation types and integrated visualization of multidisciplinary are necessary.

As mentioned above, the intermediate model is critical for multidisciplinary integrated visualization, some research has been conducted in this area. Choi et al (2002) initially proposed the feature-based multi-resolution modelling of solids. A conventional solid data structure is used as a topological structure for representing multi-resolution solid models. The lowest resolution model is made by uniting all the additive features while ones with higher resolution are generated by applying subtractive features successively in the descending order of volumes. However, this approach is computational expensive and does not allow an arbitrary rearrangement of additive or subtractive features. Lee (2005) proposed a CAD-CAE integration approach using multi-solution and multi-abstraction modeling techniques and a single master model. For the given LODs and LOAs, appropriate design and analysis models are extracted from the master model, which are stored as a non-manifold topology (NMT) (Lee et al. 2002; 2004; 2005). However, it is mainly focused on CAD and FEA-based strength analysis. Gujarathi et al. (2010; 2011) presented a CAD/CAE integration method using a Common Data Model (CDM) that contains all the required parametric information for both CAD modeling and CAE analysis. CDM, through a programmed design management structure, is connected with the design models and expert knowledge for any KBE implementation. Parametric integration of CAD and CAE using a CDM enables solving the problem of association of feature based semantic knowledge and iteration in the CAD and

CAE interaction cycles. The CDM gives the flexibility of using various CAD and CAE software tools by using a neutral data structure. However, a knowledge-based software tool including part template library needs to be developed to deal with more diverse design problems. Hamri and Drieux (Hamri et al. 2008; Drieux et al. 2007; Hamri et al. 2010) proposed a software environment for CAD/CAE integration by introducing a mixed shape representation. It mainly consists of a B-Rep NURBS topology, a faceted model and a high level topology (HLT) that represents a common requirement for the preparation of simulation model. Their approach improves the robustness of the various processes involved in FEA model preparation from CAD data and makes the overall conversion more efficient. Smit and Bronsvort (2009) proposed to add an analysis view to a multi-view feature modeling system to improve the integration of CAD and CAE models. Further research is ongoing on the link between the design and analysis models, which aims to maintain these models simultaneously and achieve information consistency within them. While all the work discussed above are more focused on automatic CAD/CAE integration or single domain visualization, little work has been done on the integration and efficient visualization of multidisciplinary simulation data despite its importance in making simulation results more accessible and useful. This research is precisely motivated by this gap and aims to develop a methodology for high-fidelity visualization of multidisciplinary simulation data.

3. Motivation and Methodology

3.1. Motivation

An easy means of obtaining the uniform intermediate mesh model is by directly constructing meshing based on the initial CAD model. However, there are some deficiencies for the directly-constructed mesh model. The first is that accuracy of visualization cannot be ensured. For multidisciplinary simulation, different critical regions actually exist. The meshes in the critical regions will be much denser than those of other regions. To realize high-fidelity visualization of

the simulation data based on the uniform intermediate mesh model, the critical regions for different disciplines cannot be ignored and must be considered carefully. The second is that the mesh model from the initial CAD model is not simple enough. The CAD model often needs to be simplified before mesh construction for multidisciplinary simulation. Generally, the simplified model after dimensional reduction and detail removal is more efficient as it provides the simulation results more rapidly without significant loss of accuracy. Furthermore, the simplification strategies are also changed for different disciplines. Therefore, a uniform simplified model should be generated for the high-fidelity and efficient visualization of multidisciplinary simulation data by removing the unnecessary details. The unnecessary details will distract designer's attention and sometimes may even reduce the global accuracy of simulation results. The third is that efficiency of visualization will be decreased when the mesh size becomes larger. As such visualization efficiency and accuracy should be considered at the same time. Based on the above analysis, the uniform intermediate model is critical and should satisfy the following requirements: it should be as simple as possible without losing any critical information of the simulation results to improve efficiency of visualization. In addition, adequate detail information should be preserved to ensure high-fidelity visualization of multidisciplinary simulation data.

3.2. Methodology

As mentioned above, it is difficult to visualize simulation results on the same platform and on the uniform model due to the heterogeneity of simulation data generated for different disciplines. To address this issue, an optimization-based method is proposed for the generation of a uniform intermediate mesh model in this study. The simulation data generated for multiple disciplines are translated into the uniform intermediate mesh model for visualization. This method aims to achieve high-fidelity and efficient visualization of multidisciplinary heterogeneous simulation data for complex product development as well as to support designers to view the simulation

results on different design platforms efficiently and accurately. The methodology of this work is shown in Figure 1, which contains three main steps:

Step 1: Formulation of the optimization problem of the uniform intermediate mesh model. In order to support high-fidelity and efficient visualization of multidisciplinary simulation results, the generation and construction of a uniform intermediate mesh model is converted to an optimization problem.

Step 2: GA-based optimization of the uniform intermediate mesh model. Based on characteristic analysis of the optimization problem, the GA method is adopted to optimize the uniform intermediate mesh model globally and locally to obtain the final ideal uniform intermediate mesh model. To refine the optimization process, a hybrid mesh size control mechanism is proposed to generate the R-initial intermediate mesh model for the optimization problem, which is computed by synthesizing the mesh size fields of simulation mesh models for different disciplines.

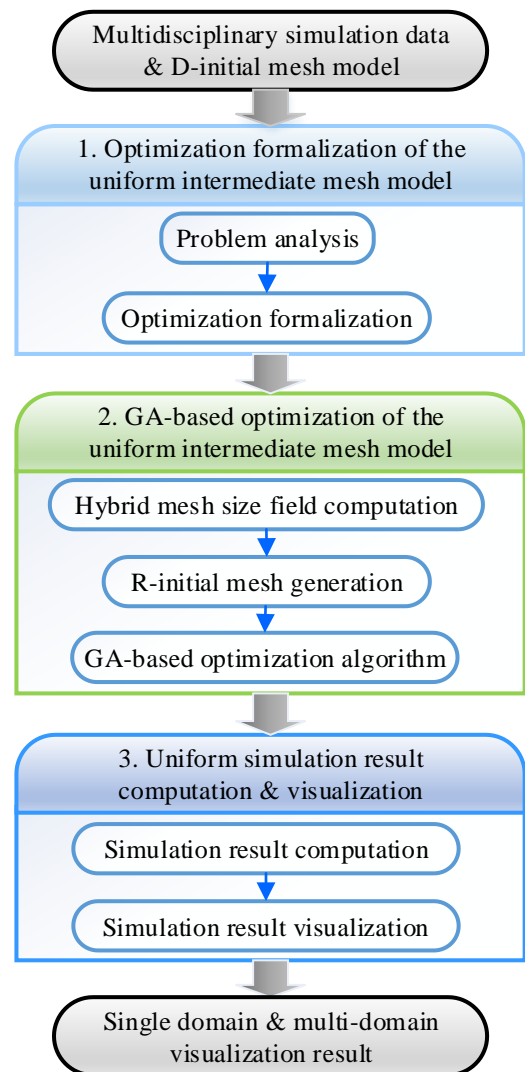


Figure 1. Methodology of research

Step 3: Computation and visualization of the simulation results based on the uniform intermediate mesh model. High-fidelity visualization of multidisciplinary heterogeneous simulation results in a CAD environment is achieved through establishing the effective mapping between the uniform intermediate mesh model data and the simulation mesh models data for different disciplines as well as the accurate interpolation of these data.

It is noteworthy that the intermediate mesh model is used for achieving the multidisciplinary visualization rather than for a specific type of simulation. Hence, the focus of this work is not on its influence on the computational accuracy and efficiency of a specific type of simulation, but on the fidelity of multiple simulation results. This means that the mesh

type, size and quantity must be considered to fully reflect the visualization requirements for different types of simulations, especially their critical regions. Based on the above analysis, the tetrahedral mesh is adopted for the uniform intermediate model. The reason for this is two-fold. The first is that the generation methods for tetrahedral mesh are relatively mature, convenient and fast. The second is that the tetrahedral mesh can deal with more complicated models with arbitrary geometry and topology.

4. Formulation of the Optimization Problem for Uniform Intermediate Mesh Model Generation

4.1. Problem analysis

As discussed in Section 3, the uniform intermediate mesh model cannot be directly obtained by conducting meshing on the uniform simplified model and cannot be too dense with its size less than those of all the simulation mesh models. In order to ensure the accuracy and efficiency of visualization, the simulation mesh models for different disciplines must be taken account of all together during the generation of the uniform intermediate mesh model. To address this issue, an optimization-based method is proposed to generate the final uniform intermediate mesh model.

Due to the fact that a model's mesh information can to some extent be reflected by its mesh size field (Zhu et al. 2002; Persson 2006; Quadros et al. 2004; Ruzi et al. 2014), its mesh distribution can be determined by its mesh size. Therefore, the hybrid mesh size field that is computed by considering all the mesh size fields of different simulation mesh models is introduced in this work to describe the mesh distribution of the intermediate mesh model. The minimum values of all the node sizes at the same position are taken as the final hybrid node sizes when computation of the hybrid mesh size field is conducted. Similarly, if the mesh size of the intermediate mesh model is less than the hybrid mesh size, this may mean the mesh is too dense. In this case, selection of such a small size is feasible but unnecessary. Actually, denser meshes do not necessarily mean higher accuracy and efficiency. In most cases, the element size can vary according to its position and function. The more

critical the region is, the finer mesh is needed.

Based on the above analysis, a cluster-based method is proposed to optimize the uniform intermediate mesh model. Specifically, the nodes are firstly clustered into K classes based on the hybrid mesh size field. The size interval of class i is marked as $[K_{i_{min}}, K_{i_{max}}]$. Meanwhile, a weight ω_i is assigned to the related node size interval to define the specific refinement degree at the corresponding position. For each node in the uniform intermediate mesh model, ω_i relative to the corresponding hybrid node size L_{mix} is firstly determined. Then the magnitude relationship between the intermediate node size L_{avg} and $\omega_i * L_{mix}$ is analyzed to determine whether further refinement is required or not. The position $\varphi_i * L_{max}$ of the longest edge L_{max} connected to the current intermediate node is taken as a refinement position during the refinement process. Moreover, a local adjustment strategy is devised, which locally adjusts the positions of the intermediate mesh nodes based on multiple simulation mesh models. After that, the uniform intermediate mesh model is adjusted step by step based on the information of the hybrid mesh size field and the simulation mesh models. In this way, approximations of the mesh models for multiple physical fields can be conducted simultaneously while all their mesh size requirements can be met with the minimum number of nodes.

4.2. Problem formulation

Suppose M is an intermediate mesh model and mark the simulation mesh models as A_i ($i = 1, 2, \dots, n$) and the hybrid mesh size field as D_{mix} . Furthermore, mark the optimized mesh of the initial mesh M as M' with the node number m and mark its corresponding mesh size field as D' . The

optimization problem of the uniform intermediate mesh model can be formulated as follows:

For $X = [\omega_1, \omega_2, \dots, \omega_K]$

$$\min F(X) = \min[f_1(X), f_2(X), f_3(X)] \quad (1)$$

$$f_1(X) = \|D' - D_{mix}\| = \sum_{j=1}^m |p_j - q_j|$$

$$f_2(X) = S = m$$

$$f_3(X) = \sum_{i=1}^n \|A_i - D'\| = \sum_{i=1}^n \left(\sum_{j=1}^{k_i} \min |o_j - p_j| \right)$$

s.t.

$$1 \leq \omega_i \leq 1.5 \quad (1 \leq i \leq K)$$

In the above equation, each X corresponds to an intermediate mesh model. $F(X)$ represents the optimization objective and S represents the minimum node number of M' . p_j , q_j and o_j are the node sizes of the j -th node in the D' , D_{mix} and the mesh size field of a simulation mesh model, respectively. Moreover, the first optimization objective $f_1(\bullet)$ is used to minimize the node size deviation between the mesh nodes and discrete nodes of the intermediate mesh model and their equivalents in the hybrid mesh size field. The second optimization objective $f_2(\bullet)$ is used to minimize the number of intermediate mesh nodes. And the third optimization objective $f_3(\bullet)$ is used to minimize the nearest distance between the intermediate mesh nodes and the heterogeneous simulation mesh nodes.

5. GA-based Optimization of the Uniform Intermediate Mesh Model

It can be seen from the problem formulation that the optimization of the uniform intermediate mesh

model is a multi-objective problem. Much research has been conducted on multi-objective optimization (MOO) [Fonseca and Fleming 1995; Marler and Arora 2004; Jin et al. 2001; Deb et al. 2002] problems, which can be classified into two categories. One involves combining all of the objective functions into a single composite function or considers all but one objective as constraint conditions. Thus, the algorithms for single-objective problems can be adopted. The other is focused on determining a Pareto front or a representative subset that does not dominate other functions. With the consideration that there are many mature single-objective optimization algorithms that are readily available, the MOO problem in this study is transformed to a single objective by weighted sum of the above three objectives.

Many optimization algorithms have been developed, including the traditional algorithms such as simplex method, Powell method and penalty function as well as the heuristic-based algorithms such as simulated annealing (SA), genetic algorithm (GA), particle swarm optimization (PSO) and neural networks (NNs) [Schaffer et al. 1992; Sun et al. 2012; Wei and Qiqiang 2004]. Although the traditional gradient-based optimization algorithms are widely used and have many advantages such as high efficiency and reliability, GA is adopted in this work. The main reason is that it is a stochastic searching algorithm and thus there are no special requirements on the continuity and convexity of the objective function and constraints. Moreover, it proves to be a very good algorithm in terms of computational complexity and global search capability [Kumar et al. 2010; Sivaraj and Ravichandran 2011].

5.1. GA-based optimization process

To optimize the uniform intermediate mesh model using the GA-based method, the first task is to transform the optimization problem discussed in Section 4 into a GA problem. In this study, a weight value ω_i is used to represent a gene and each individual (corresponding to an intermediate mesh model) consists of a set of weights. The values of the

objective functions are used to construct the fitness function F as follows:

$$F = 1/(\alpha S^* + \beta V^* + \gamma T^*) \quad (2)$$

Specifically, S^* represents the normalized number of the intermediate mesh nodes; V^* is the normalized absolute value of node size deviation between the mesh nodes and discrete nodes of the intermediate mesh model and their equivalents in the hybrid mesh size field; and T^* is the normalized sum

of the nearest distance between the intermediate mesh nodes and the simulation mesh nodes. For each individual, its key parameters are computed as follows:

$$\begin{cases} S_i^* = (S_i - S_{min})/(S_{max} - S_{min}) \\ V_i^* = (V_i - V_{min})/(V_{max} - V_{min}) \\ T_i^* = (T_i - T_{min})/(T_{max} - T_{min}) \end{cases} \quad (3)$$

In Equation (3), S_i^* , V_i^* and T_i^* represent the normalized values of S_i , V_i and T_i of the i -th individual, respectively. S_{max} , S_{min} , V_{max} , V_{min} and

T_{max} , T_{min} denote the maximum and minimum values of the corresponding sample population. According to the definition of the fitness function, the greater the fitness value, the better the solution.

Furthermore, some other parameters such as the number of the initial population, the crossover probability (P_c), mutation probability (P_m) and the maximum number of iterations are set in accordance with the algorithm's specific needs. It is noteworthy that all of the parameters for different designers can be altered later on. Once the genetic representation and the fitness function are defined, the GA process initializes a population of individuals and then improves it through repetitive applications of selection, mutation and crossover operators. Moreover, a further local adjustment mechanism is needed to adjust the

positions of the intermediate mesh nodes. This mechanism can minimize the nearest distance between the intermediate mesh nodes and heterogeneous simulation mesh nodes, which improves the accuracy of data mapping. Therefore, the local adjustment rule of an intermediate mesh node is to minimize the sum of its distance from the corresponding simulation mesh nodes. Through the above process, an optimal uniform intermediate mesh model which satisfies the predefined optimization objectives is obtained to facilitate the subsequent high-fidelity and efficient visualization of multidisciplinary heterogeneous simulation results.

5.2. Improvement of GA for optimization of the uniform intermediate mesh model

Generally, the selection of P_c and P_m is critical to the performance and convergence of GA. The larger the value of P_c , the faster a new individual is generated. However, the individuals with higher fitness values may soon be excluded as well. Moreover, the search process will be slow if the value of P_c is too small. On the other hand, it is difficult to generate a new individual with a small P_m . In addition, the GA process becomes a random search algorithm if P_m is too large. To overcome this problem, Srinivas and Patnaik (1994) proposed an adaptive genetic algorithm (AGA). The parameters P_c and P_m are adaptively changed according to the fitness using the following equations:

$$P_c = \begin{cases} \frac{k_1(f_{max} - f')}{f_{max} - f_{avg}}, & f' \geq f_{avg} \\ k_2, & f' < f_{avg} \end{cases} \quad (4)$$

$$P_m = \begin{cases} \frac{k_3(f_{max} - f')}{f_{max} - f_{avg}}, & f' \geq f_{avg} \\ k_4, & f' < f_{avg} \end{cases} \quad (5)$$

Specifically, f_{max} represents the maximum

fitness value of the population; f_{avg} represents the average fitness value of each generation of the population; f' is the larger fitness value of the two crossover individuals; f is the fitness value of the individual to be mutated; and k_i ($i = 1-4$) are constant.

The basic idea of AGA is that for the individuals with their fitness values higher than the average, a lower P_c and P_m are adopted to protect the individuals with excellent performance so they can be included in the next generation. On the contrary, for the individuals with their fitness lower than the average, a higher P_c and P_m are adopted to exclude the poor performance individuals as much as possible. However, there still exists a problem, that is, P_c and P_m are smaller when the individual fitness is closer to the maximum fitness. In this sense, the above adaptive method is more appropriate for the later evolution phase of the population whereas it is not suitable for the earlier evolution phase. The reason is that the individuals with a high fitness and a small P_c lead to a slow evolution since the higher fitness individuals almost does not change or has little change, especially under the condition of a large number of individuals with higher fitness.

Based on the above analysis, a gradient and mutation mechanism in the biological evolution process is introduced in this study to improve the traditional AGA. The process of implementing the improvements are given as follows:

- (1) Sort all the individuals of the population according to their fitness values. The population is divided into two groups, namely the gradient population with higher fitness and the mutant population with lower fitness.
- (2) Compute the maximum fitness values and average

fitness values of the gradient population and the mutant population, respectively. Then, the AGA is adopted for each population.

Using the above improved AGA with the gradient and mutation mechanism, the values of P_c and P_m of excellent individuals can be improved to ensure the diversity of the entire population. Furthermore, the improved AGA can prevent the searching process from falling into a local optimum so as to get a global optimal solution with a higher rate of convergence.

6. Generation of the R-Initial Intermediate Mesh Model

The initial population affects the performance of GA greatly. A good initial population can improve the algorithm's performance. As mentioned above, the direct initial (D-initial) intermediate mesh model is obtained by conducting meshing on the uniform intermediate simplified model. It is noteworthy that the simulation mesh models of multiple disciplines provide important heuristics for the final optimal uniform intermediate mesh model. Based on this observation, a method based on a hybrid mesh size field with the consideration of multiple simulation mesh models is proposed to obtain the R-initial intermediate mesh model as the input for the GA-based optimization.

6.1. Computation of the hybrid mesh size field

The mesh size refers to the number of the elements of the whole or partial area of the finite element mesh, which determines the solving space of FEA and eventually influences the computational efficiency and accuracy of simulation. In most cases, the mesh size function is used to describe mesh distribution, such as a scale function or a Riemannian metric field. Specifically, the mesh size of a specific node is defined by averaging its edges:

$$L_{avg}(P) = (\sum_{i=1}^N L_i)/N \quad (6)$$

In Equation (6), $L_{avg}(P)$ is the size for node P ,

N is the number of edges connecting to P , and L_i is

the length of the i -th edge connecting to P .

According to Equation (6), the mesh size of all nodes can be computed and the whole mesh size field of the simulation mesh model of each discipline can be obtained as well. Then, a method to compute the hybrid mesh size field based on those of multiple simulation mesh models becomes very important for obtaining the R-initial mesh model, which is formalized as follows:

Given the mesh size field $h_i(x), i = 1, 2, \dots, n$, a new mesh size field can be defined in Equation (7):

$$h_{mix}(x) = \min_{i=1, \dots, n}(h_i(x)) \quad (7)$$

As shown in the equation, $h_{mix}(x)$ equals to the minimum of all the node sizes as the hybrid node size need satisfy different size control requirements at the same position. According to the above definition, the hybrid mesh size field can be constructed through the following three steps, as shown in Figure 2:

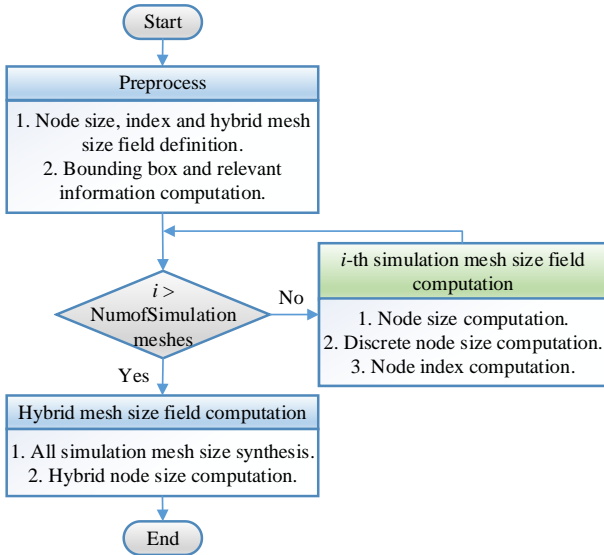


Figure 2. The flowchart of hybrid mesh size field computation.

Step 1: Compute the bounding box of the intermediate simplified model and obtain its lower left corner $(X_{min}, Y_{min}, Z_{min})$ and its upper right corner $(X_{max}, Y_{max}, Z_{max})$. For each element of all the

simulation mesh models, take the smallest length $GriCellSize$ as a unit to create the indexes of the intermediate simplified model. Therefore, the size of the index matrix $GriCellIndex[X_{size}][Y_{size}][Z_{size}]$ is:

$$\begin{cases} X_{size} = (X_{max} - X_{min})/GriCellSize + 1 \\ Y_{size} = (Y_{max} - Y_{min})/GriCellSize + 1 \\ Z_{size} = (Z_{max} - Z_{min})/GriCellSize + 1 \end{cases} \quad (8)$$

In addition, for a specific node $P(X, Y, Z)$, its corresponding index is computed using the following equation:

$$\begin{cases} i = (X - X_{min})/GriCellSize \\ j = (Y - Y_{min})/GriCellSize \\ k = (Z - Z_{min})/GriCellSize \end{cases} \quad (9)$$

Step 2: For each simulation mesh model, its corresponding mesh size field can be constructed as follows:

- Mesh node size computation. For each node in the mesh model, compute its size based using Equation (6).
- Discrete node size computation. For each element in the mesh model, take the cell size $GriCellSize$ as a unit to conduct discretization. Suppose the longest edge of an element concerned is L_{max} , then the discrete number of segments Sag is:

$$Sag = L_{max}/GriCellSize + 1 \quad (10)$$

After that, the discretized node size is computed based on the barycentric coordinate equation (Shao et al. 2013).

- Node index computation. Compute the node index using Equation (9). Set the node size and index information for the simulation mesh model based on all the node information including the mesh nodes and discrete nodes.

Step 3: Compute the hybrid mesh size field by synthesizing the simulation mesh size fields of different disciplines. For each node in the hybrid mesh size field, obtain the final hybrid node size using

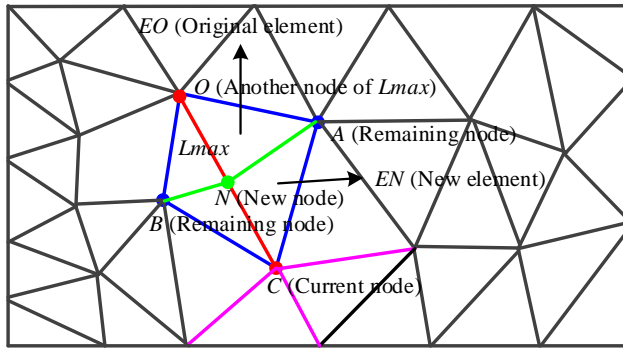
Equation (7).

6.2. R-Initial mesh generation based on hybrid mesh size control

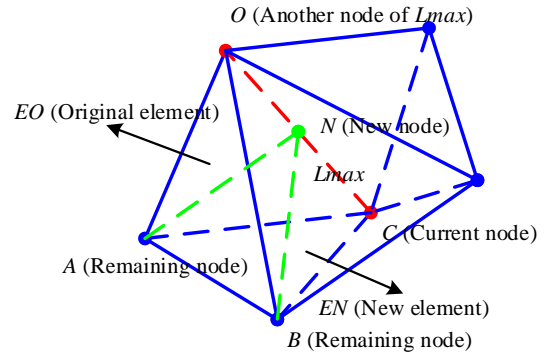
Generally, mesh size control is used to ensure that the mesh size concerned satisfies the requirements of the predefined mesh size function by adding or deleting elements, which is mainly comprised of mesh coarsening and refinement to minimize the size deviation between the current mesh model and the ideal mesh model. In this work, the R-initial intermediate mesh generation based on hybrid mesh size control is developed to change the topology and geometry of the D-initial intermediate mesh model through mesh refinement to satisfy the hybrid mesh

size field while minimizing mesh size deviation. Ideally, the R-initial intermediate mesh model should approximate multiple simulation mesh models as much as possible while being able to maintaining all their mesh sizes especially the critical regions.

Based on the above analysis, the algorithm for R-initial intermediate mesh generation is conducted if L_{avg} is larger than $\omega_i * L_{mix}$. As shown in Figure 3, for the triangular mesh and the tetrahedral mesh, the position $\varphi_i * L_{max}$ of the longest edge L_{max} connected to the current node C is taken as a refinement position. The mesh refinement algorithm is described as follows:



(a) Triangular mesh adjustment



(b) Tetrahedral mesh adjustment

Figure 3. Illustration of mesh refinement and information update.

- Create a new mesh node N at the refinement position, as shown in Figure 3a.
- Find the common elements that contain L_{max} . For each common element, find the remaining nodes that do not belong to L_{max} .
- Update the information of all the common elements. Split the original element into two by connecting the remaining nodes. As shown in Figure 3a, the remaining node is A or B . In Figure 3b, they are both A and B . After being split, the original element EO is composed of the remaining nodes, the new node N and another node O of L_{max} , while the new element EN contains the

remaining nodes, the new node N and the current node C .

- Update the node information of all the split elements, especially the elements list (the elements that contains the current node C). Add the new element EN to the remaining nodes. Keep another node O in L_{max} unchanged. Delete the original element EO and add the new element EN to the element list of the current node C . In addition, the new node includes the original element EO and the new element EN .
- Update relevant information of the node size by re-computing the node size of all the updated elements.

The above process will be iterated until the uniform intermediate mesh model satisfies the requirement of the hybrid mesh size field. Taking the 2D surface in the *cabinet* part as an example, the R-initial intermediate mesh model generation process is illustrated in Figure 4. Compared with D-initial

intermediate mesh model, two features (hole 1 and hole 2) in simulation mesh A and two features (hole 3 and hole 4) in simulation mesh B are refined, respectively. Thus, hole 1, hole 2, hole 3 and hole 4 in the R-initial intermediate mesh model are all refined using the hybrid mesh size control method.

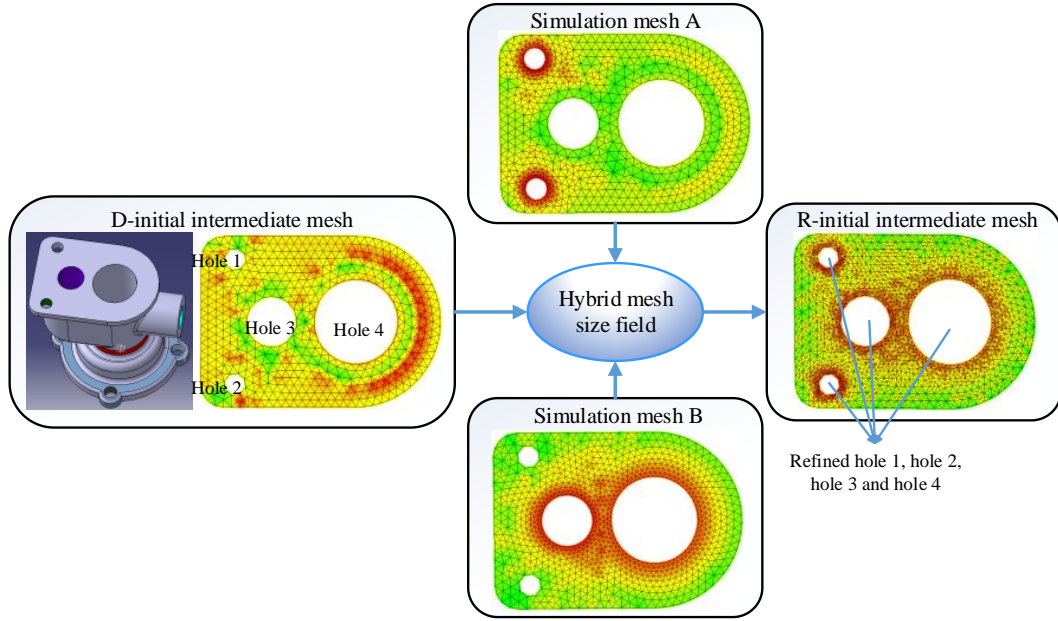


Figure 4. R-initial intermediate mesh model generation.

7. Efficient Computation and Evaluation of the Uniform Intermediate Simulation Data

After the uniform intermediate mesh model is optimized, a mechanism for data mapping and interpolation should be established to transfer information between the uniform intermediate mesh model and multiple simulation mesh models effectively and efficiently. Furthermore, in order to speed up the computation process, indexing and parallel computing strategies are devised. The whole process is shown in Figure 5.

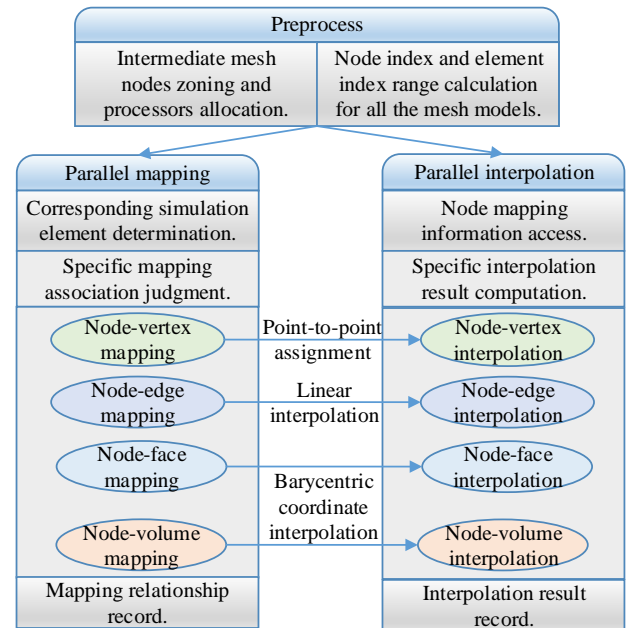


Figure 5. The process of the uniform intermediate simulation results computation.

7.1. Data mapping and interpolation

Simulation results of the uniform intermediate mesh model can be divided into the external and internal parts. Since all the simulation results are discrete values on mesh nodes, the external simulation results are those on the vertex nodes, edge nodes and face nodes while the internal simulation results are those on the internal nodes. Therefore, according to spatial coordinates and topological information, data mapping can be done by establishing associations between the intermediate nodes and the mesh nodes, elements, faces or volumes of different simulation mesh models. Since the mapping between model surfaces is a one-to-one mapping, the external mapping relationships can be divided into three kinds, namely node-vertex mapping, node-edge mapping and node-face mapping, while the internal mapping relationships including four specific sets. In addition to the three types of mapping mentioned above, another type is node-volume mapping.

Based on the above analysis, the entire mesh mapping algorithm can be realized by analyzing the specific spatial relationships between the intermediate mesh nodes and those of the corresponding simulation mesh elements. If the distance between the intermediate node and one of the element nodes is less than a predefined threshold, it is a node-vertex mapping. Otherwise, the spacial position (on an edge, on a face or in the element) of the node should be determined. For the three specific special positions, the mapping type will be node-edge mapping, node-face mapping or node-volume mapping, respectively.

After that, the next task is to obtain the simulation data values of the intermediate mesh nodes based on the mapping types identified. A data interpolation method is used to obtain the simulation results for the uniform intermediate mesh model (Shao et al. 2013). Corresponding to different mapping relationships, four kinds of interpolation strategies are employed in this study, namely node-vertex interpolation, node-edge interpolation, node-face interpolation and node-volume interpolation. Specifically, the node-vertex interpolation is implemented by point-to-point assignment; the node-edge interpolation is by linear

interpolation; and both the node-face and the node-volume interpolation are by barycentric coordinate interpolation.

7.2. Strategies to improve computational efficiency

Since the number of the mesh nodes and elements are huge for complex product development, two strategies are particularly proposed to improve the computational efficiency. One is the indexing of mesh nodes as mentioned in Section 6.1, which can be used to narrow down the element detection range to improve computational efficiency during the mapping process. The other is the parallel computing strategy that emphasizes conducting all the node mapping and interpolation computation in separate processors. Specifically, all the nodes are divided into a certain number of groups according to the number of processors firstly. Then, each group of the nodes is assigned a processor. Finally, a parallel computing strategy is used to establish the mapping and interpolation relationships to quickly obtain the corresponding physical field values of the intermediate mesh nodes.

7.3. Performance evaluation

In order to demonstrate the performance of the GA-based uniform intermediate mesh model, it is necessary to evaluate the visualization accuracy by computing the Error Percentage (EP) between the actual value and the expected one for each node of the simulation mesh model as follows:

$$EP = \frac{|(V_{actual} - V_{expected})|}{V_{actual}} \times 100\% \quad (11)$$

In Equation (11), V_{actual} and $V_{expected}$ denote the actual and the expected value of each simulation node. Generally, the lower the value of EP , the more accurate the GA-based uniform intermediate mesh model. Since the V_{actual} of the simulation mesh model has been obtained, the next task is to compute

the value of $V_{expected}$. In this work, it is computed by

conducting reverse mapping and interpolation from the uniform intermediate mesh model to all simulation models for different disciplines. Therefore, its computation process is similar to the one described in Section 7.1.

With the consideration that the number of mesh nodes are huge for complex product development, the Statistical Error Percentage (SEP) is adopted here as the evaluation criterion. For the i -th given error interval $[EPi_{min}, EPi_{max}]$, its SEP is computed as follows:

$$SEP_i = \sum_{j=1}^k g(EP_j) / k \quad (12)$$

If $EP_j \in [EPi_{min}, EPi_{max}]$, $g(EP_j)$ equals 1; otherwise, it is 0. k represents the node number of the corresponding simulation mesh model. The error intervals of the statistical results are decided depending on the SEPs of all the mesh nodes and the particular simulation type. It is noteworthy that the node error percentage of structural field analysis in Abaqus is relatively larger than that of other types of simulation when applying this evaluation criterion, due to the fact

that stress is distributed on elements.

8. Experiments and Discussion

To evaluate the proposed models and methods, a prototypical system is implemented based on the CATIA CAD platform and all the programming work is done using Microsoft visual studio 2008. Two commercial CAE systems are chosen for a typical multi-field simulation, namely Abaqus for the structural field analysis and ANSYS for the thermal field analysis. To demonstrate the performance of the proposed uniform intermediate model, four groups of experiments are conducted as follows. The first is used to demonstrate the whole process of applying the proposed method. The second is about the influence of the optimization parameters in the GA-based optimization process on the performance of model generation and visualization. The third is about performance analysis including error comparison and evaluation of the interpolated simulation results on different intermediate mesh models, the corresponding time cost and visualization results. The last experiment is the efficiency analysis of the measure of the relationship between the intermediate mesh node number and the visualization accuracy and efficiency.

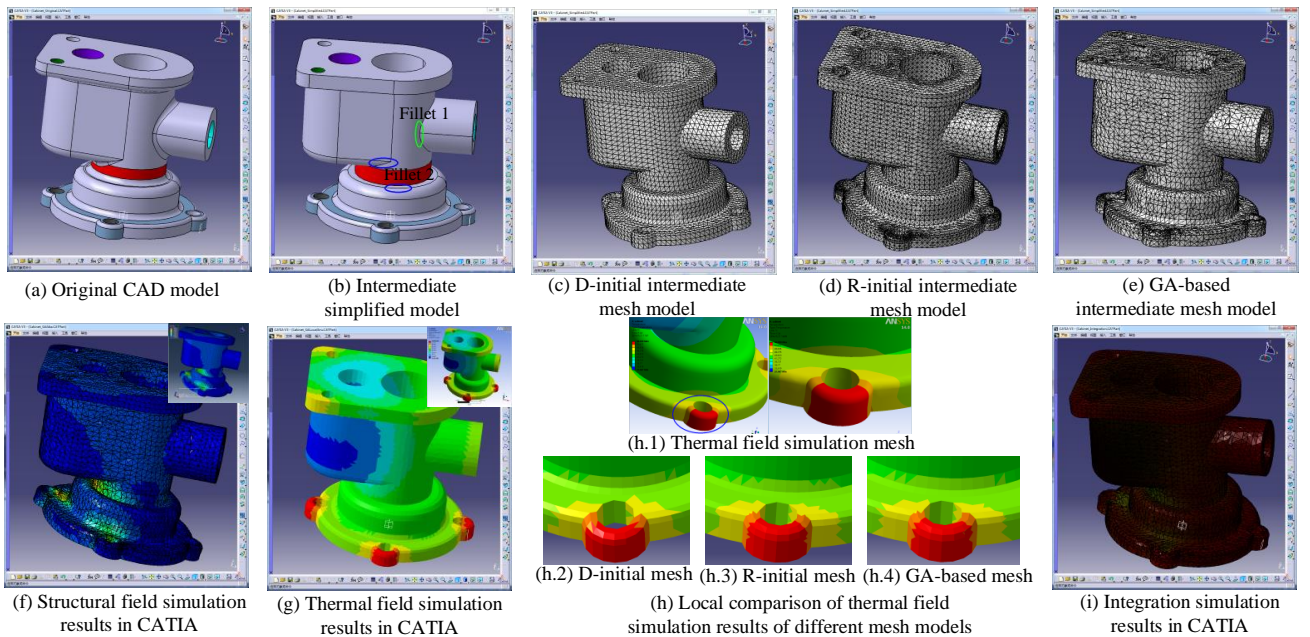


Figure 6. The process of uniform intermediate mesh model generation and visualization.

8.1. Demonstration of the whole intermediate mesh model generation and visualization process

Figure 6 shows the whole process of the generation of uniform intermediate mesh models and the high-fidelity and efficient visualization of multidisciplinary simulation data. Specifically, Figure 6a shows the original CAD model of a part named *cabinet*. The uniform intermediate simplified model with two common simplification features, namely fillet 1 and fillet 2, is shown in Figure 6b. Then the D-initial intermediate mesh model is obtained by conducting meshing directly on the CAD model, as shown in Figure 6c. Next, the R-initial intermediate mesh model, as shown in Figure 6d, is generated based on the hybrid mesh size control method. After that, the optimized uniform intermediate mesh model is obtained after the GA-based optimization, as shown in Figure 6e. Finally, the mapping and interpolation processing is conducted to transfer data between the uniform intermediate mesh model and those generated by Abaqus and ANSYS. The simulation results

obtained using these two packages are visualized in CATIA in Figure 6f and Figure 6g, respectively. Particularly, the local comparison of thermal field simulation results for different mesh models is shown in Figure 6h. It can be seen from Figure 6 that the simulation results in Figure 6(h.3) and Figure 6(h.4) get are quite similar. Moreover, these results are much better than those obtained using the D-initial intermediate mesh model as shown in Figure 6(h.2). The specific simulation data and statistical results of the local mesh will be given in Section 8.3. In this case, the GA-based uniform intermediate mesh model has higher visualization accuracy and efficiency with fewer nodes, and more details will be given in Section 8.2. Integrated visualization in CATIA is shown in Figure 6i where red is used to represent temperature change of the thermal field simulation in ANSYS and green is used to display the stress change of the structural field simulation in Abaqus. Meanwhile, blue is set to 0 by default.

Table 1. Results of orthogonal experiments with different combinations of optimization parameters K , α , β and γ .

No	K	(α, β, γ)	Node number	Time/s		SEP in Abaqus				SEP in ANSYS				Average SEP
				TABA	TANS	0~5%	5%~10%	10%~20%	>20%	0~0.3%	0.3%~0.6%	0.6%~0.9%	>0.9%	
1	1	(0.25,0.25,0.5)	9713	1.888	1.654	83.76%	10.92%	3.79%	1.53%	86.32%	12.18%	1.11%	0.38%	85.04%
2	1	(0.3,0.3,0.4)	11639	2.262	2.137	85.71%	9.76%	3.35%	1.18%	89.07%	9.78%	0.79%	0.35%	87.39%
3	1	(0.35,0.35,0.3)	11557	2.169	2.09	85.62%	9.79%	3.37%	1.21%	88.97%	9.83%	0.85%	0.35%	87.29%
4	1	(0.4,0.4,0.2)	11033	2.09	1.997	85.17%	10.00%	3.54%	1.29%	88.53%	10.28%	0.92%	0.28%	86.85%
5	1	(0.45,0.45,0.1)	11084	2.138	1.965	85.22%	9.94%	3.57%	1.28%	88.50%	10.29%	0.93%	0.29%	86.86%
6	2	(0.25,0.3,0.45)	12448	2.356	2.277	86.72%	9.24%	3.02%	1.02%	88.58%	10.26%	0.87%	0.29%	87.65%
7	2	(0.3,0.35,0.35)	13278	2.506	2.618	87.50%	8.78%	2.76%	0.96%	89.71%	9.34%	0.71%	0.24%	88.61%
8	2	(0.35,0.4,0.25)	12420	2.308	2.324	86.57%	9.39%	3.04%	1.01%	89.48%	9.42%	0.80%	0.30%	88.02%
9	2	(0.4,0.45,0.15)	11494	2.09	2.215	85.77%	9.95%	3.28%	1.01%	87.33%	11.27%	1.09%	0.31%	86.55%
10	2	(0.45,0.25,0.3)	11497	2.121	2.028	85.84%	9.84%	3.20%	1.12%	88.14%	10.61%	0.90%	0.34%	86.99%
11	3	(0.25,0.35,0.4)	10789	2.06	1.903	84.91%	10.20%	3.59%	1.30%	87.49%	11.22%	0.89%	0.40%	86.20%
12	3	(0.3,0.4,0.3)	12045	2.355	2.215	86.34%	9.39%	3.23%	1.05%	88.98%	9.91%	0.76%	0.36%	87.66%
13	3	(0.35,0.45,0.2)	11656	2.308	2.184	85.78%	9.69%	3.39%	1.14%	88.81%	10.02%	0.85%	0.32%	87.29%
14	3	(0.4,0.25,0.35)	11067	2.153	2.043	85.14%	10.17%	3.46%	1.23%	88.06%	10.68%	0.90%	0.36%	86.60%
15	3	(0.45,0.3,0.25)	10477	2.043	1.903	84.44%	10.38%	3.74%	1.45%	87.44%	11.22%	0.94%	0.40%	85.94%
16	4	(0.25,0.4,0.35)	14372	2.746	2.652	88.46%	8.25%	2.54%	0.75%	90.78%	8.35%	0.65%	0.22%	89.62%
17	4	(0.3,0.45,0.25)	12872	2.386	2.262	87.08%	9.07%	2.85%	1.00%	89.01%	10.01%	0.76%	0.22%	88.05%
18	4	(0.35,0.25,0.4)	11666	2.262	2.215	85.96%	9.68%	3.26%	1.10%	88.07%	10.64%	0.95%	0.34%	87.01%
19	4	(0.4,0.3,0.3)	10659	2.028	1.934	84.60%	10.41%	3.72%	1.27%	87.10%	11.47%	1.06%	0.36%	85.85%
20	4	(0.45,0.35,0.2)	11249	2.059	2.137	85.59%	10.06%	3.28%	1.07%	87.36%	11.33%	0.98%	0.33%	86.48%
21	5	(0.25,0.45,0.3)	13789	2.684	2.512	87.95%	8.37%	2.73%	0.94%	90.61%	8.52%	0.57%	0.30%	89.28%
22	5	(0.3,0.25,0.45)	11941	2.309	2.246	86.00%	9.56%	3.42%	1.02%	88.65%	10.15%	0.78%	0.41%	87.32%
23	5	(0.35,0.3,0.35)	10216	1.934	1.872	84.09%	10.87%	3.83%	1.21%	86.12%	12.36%	1.13%	0.38%	85.11%
24	5	(0.4,0.35,0.25)	11251	2.153	2.074	85.08%	10.16%	3.58%	1.18%	87.74%	10.93%	0.95%	0.38%	86.41%
25	5	(0.45,0.4,0.15)	11240	2.06	2.09	85.25%	9.90%	3.64%	1.21%	87.98%	10.80%	0.92%	0.30%	86.61%
26	1	D-initial mesh	5675	1.339	1.327	77.35%	15.43%	5.46%	1.75%	80.69%	17.58%	1.44%	0.30%	79.02%
27	1	R-initial mesh	20445	5.328	4.782	84.77%	10.44%	3.76%	1.03%	90.86%	0.30%	0.51%	0.04%	87.82%

Note: TABA = "Time to obtain the simulation results in Abaqus", TANS = "Time to obtain the simulation results in ANSYS".

8.2. Analysis of the optimization parameters

During the optimization process of the uniform intermediate mesh model, there are several optimization parameters, i.e. K , α , β and γ , which determine the quality of optimization results and thus must be chosen carefully. In this study, a group of orthogonal experiments are conducted and the results of different combinations of these optimization parameters are listed in Table 1. In these experiments, the part *piston* is used to obtain results for comparison. The following observations can be made from the data in Table 1:

- (1) The errors vary for different combinations of optimization parameters. Generally, the more the number of intermediate mesh nodes, the smaller the errors and the longer the time to obtain simulation results. For example, in experiment 1 the number of mesh nodes 9713 is less than the node number 14372 in experiment 16. With the increase of node number, the SEP in [0, 5%] is increased from 83.76% to 88.46% in Abaqus and the SEP in [0, 0.3%] is increased from 86.32% to 90.78% in ANSYS. However, the total mapping and interpolation processing time increases by $(2.746+2.652-1.888-1.654)/(1.888+1.654) = 52.40\%$.
- (2) The parameters K and γ have a relatively small impact on the test results while α and β are more influential. In addition, α , β and γ have some sort of dependency on each other. It can be found that smaller α and larger β are set for smaller errors after the comparison of the 8th, 12th, 16th, 17th and 21st experiments.
- (3) The GA-based optimization mesh model with less node number can achieve a similar or even better result than the R-initial mesh model. For example, in the 7th, 8th, 16th, 17th and 21st experiments GA-based mesh model have better results than the R-initial mesh model based on the average performance. Moreover, the 8th and 17th experiments take less time to obtain the simulation results than the 21st experiment due to the less number of mesh nodes. The node number (20445) of R-initial mesh model in the 27th

experiment is more than that (12420) of the 8th experiment. However, the SEP in [0, 0.5%] is increased from 84.77% to 86.57% in Abaqus and little difference is observed in [0, 0.3%] in ANSYS. The average SEP is increased from 87.82% to 88.02%. As a result, the GA-based uniform intermediate mesh model obtain a higher visualization accuracy than the R-initial mesh model with the time cost reducing by $(5.328+4.782-2.308-2.324)/(5.328+4.782) = 54.18\%$.

Therefore, the errors and the time cost are both affected by the number of mesh nodes. Appropriate parameters can be selected according to the measure of the relationship between visualization accuracy and visualization efficiency. Based on the above observations and analysis, the optimization parameter combination in the 8th experiment is selected in this study, that is, K is set to be 2, and (α, β, γ) is set to be (0.35, 0.4, 0.25).

8.3. Performance experiments and analysis

In order to demonstrate the performance of the proposed approach, several typical experiments are conducted. Specifically, the *cabinet* part, the *piston* part and the *support arm* part are selected as the test objects. Tables 2 through to4 show the statistical results of the error percentage and corresponding time cost of the above three parts.

Table 2. The SEP and time cost of the *cabinet* part.

Intermediate mesh model	Node number	Time/s		SEP		
		TAba	TAns	0~5%	0~0.2%	Average
D-initial mesh	13625	13.151	10.415	73%	77.99%	75.49%
R-initial mesh	38552	38.529	29.268	79.48%	89.26%	84.37%
GA-based mesh	25031	20.154	16.078	84.06%	89.15%	86.61%

Table 3. The SEP and time cost of the *piston* part.

Intermediate mesh model	Node number	Time/s		SEP		
		TAba	TAns	0~5%	0~0.3%	Average
D-initial mesh	5675	1.339	1.327	77.35%	80.69%	79.02%
R-initial mesh	20445	5.328	4.728	84.77%	90.86%	87.82%
GA-based mesh	12420	2.308	2.324	86.57%	89.48%	88.02%

Table 4. The SEP and time cost of the *support arm* part.

Intermediate mesh model	Node number	Time/s		SEP		
		TABA	TAns	0~5%	0~0.2%	Average
D-initial mesh	7715	4.523	1.916	81.02%	83.77%	82.39%
R-initial mesh	24360	14.234	5.216	85.24%	93.67%	89.46%
GA-based mesh	14464	6.712	3.272	88.34%	92.83%	90.58%

It can be concluded from the three tables that the visualization results of R-initial intermediate mesh model and GA-based uniform intermediate mesh model are much better than those obtained by using the D-initial intermediate mesh model. In addition, the time cost increases with the increase of the mesh node number. As a result, the less the mesh node number, the more the time that can be saved, especially for complicated models with a large number of mesh nodes. Moreover, the GA-based uniform intermediate mesh model with fewer nodes usually obtain a similar or even better statistical results than the R-initial intermediate mesh model. This phenomenon is explained as follows.

Take the *cabinet* part in Table 2 as an example, the node number of R-initial intermediate mesh model is 38552, which becomes 25031 after the GA-based optimization. There is a difference of 13521 (38552-25031). However, the SEP for the structural field analysis in Abaqus is increased from 79.48% to 84.06% in [0, 5%], and the SEP in [0, 0.2%] for the thermal field analysis in ANSYS has little difference. In this case, the GA-based uniform intermediate mesh

model achieves better performance with a higher average SEP 86.61%. Moreover, the GA-based uniform intermediate mesh model takes less time to obtain simulation results than the R-initial intermediate mesh model. The time cost of R-initial mesh model for the structural field analysis in Abaqus and the thermal field analysis in ANSYS are 38.529s and 29.268s, respectively, while those of the GA-based uniform intermediate mesh model are 20.154s and 16.078s, respectively. It means the time cost is reduced by $((38.529+29.268-20.154-16.078)/(38.529+29.268)) = 46.56\%$.

The statistical results are similar for the *piston* part in Table 3 and the *support arm* part in Table 4. For the *support arm* part, the R-initial intermediate mesh node number is 24360, while the GA-based optimized mesh model is 14464. Even though the SEP for the thermal field analysis in ANSYS is reduced from 93.67% to 92.83%, the average SEP is increased from 89.46% to 90.58%. Furthermore, the time cost to obtain simulation results is reduced by $((14.234+5.216-6.712-3.272)/(14.234+5.216)) = 48.67\%$.

Moreover, the GA-based uniform intermediate mesh model can always achieve similar or even better performance with less time cost. The specific error percentage statistics for the three parts mentioned above are given in Figures 7 through to 9.

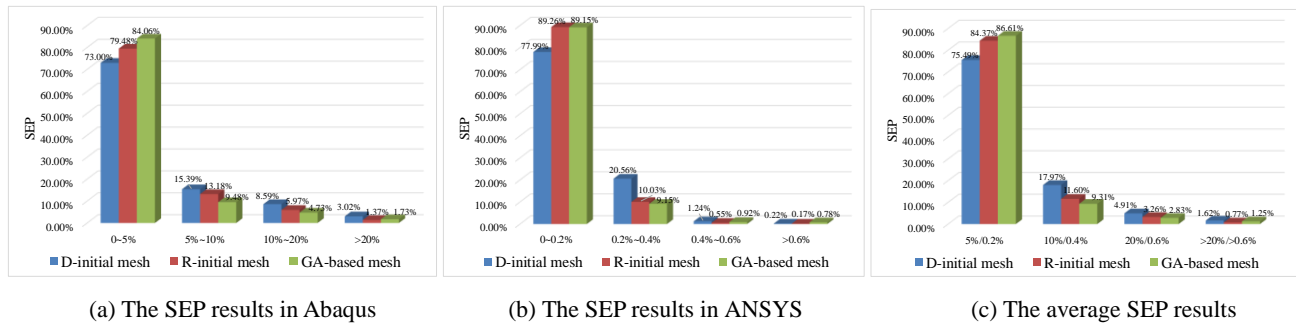


Figure 7. The SEP results of the *cabinet* part.

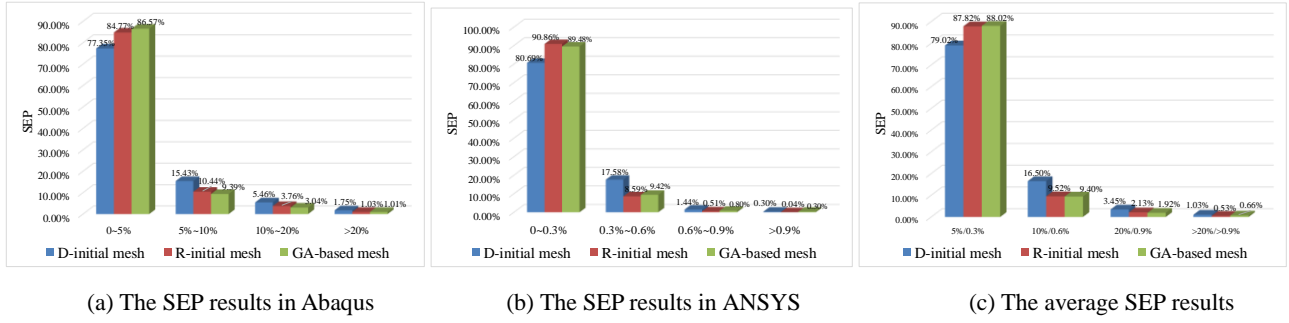


Figure 8. The SEP results of the *piston* part.

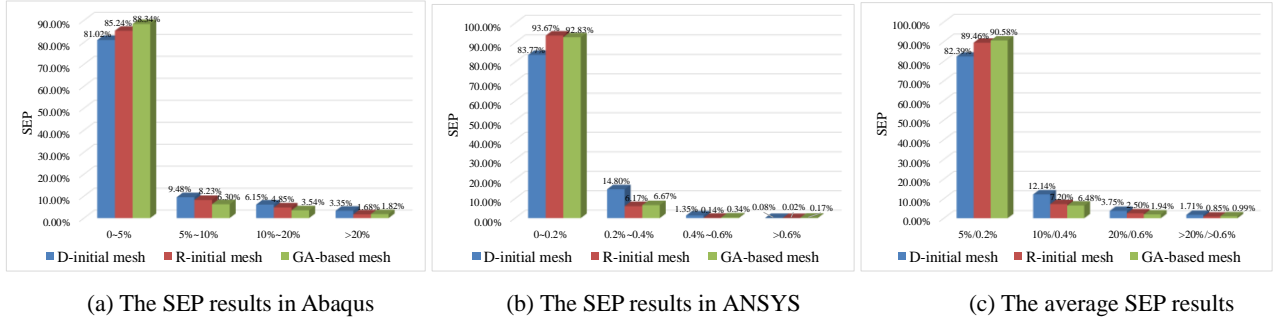


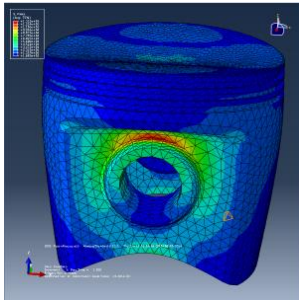
Figure 9. The SEP results of the *support arm* part.

Based on the above analysis, the GA-based uniform intermediate mesh model can significantly improve computational efficiency while ensuring or strengthening the visualization accuracy since it can largely reduce the mesh node number with the consideration of both global and local accuracy.

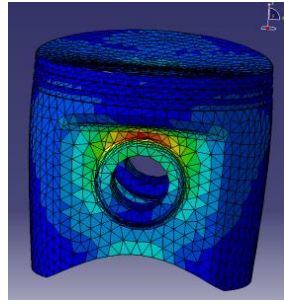
In addition to the visualization results for the *cabinet* part discussed in Section 8.1, the visualization results of the *piston* part and the *support arm* part are shown in Figures 10 through to 13. Three simulation experiments are conducted for each part using the D-initial, the R-initial and the GA-based uniform intermediate mesh model, respectively. It can be seen from the figures that the visualization accuracy of R-initial intermediate mesh model and GA-based uniform intermediate mesh model are relatively better than those of the D-initial intermediate mesh model.

To demonstrate the superiority of the GA-based uniform intermediate mesh model especially for the local critical regions, two further experiments for the structural field analysis and the thermal field analysis

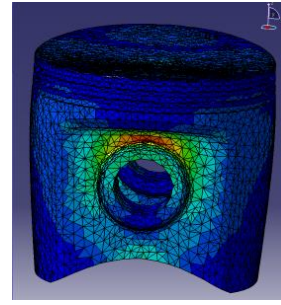
are conducted. As the local visualization results of the *cabinet* part in the thermal field analysis have been given in Figure 6h, a typical example of the structural field analysis for the *piston* part is given in Figure 14. The local statistical results of different intermediate mesh models of the above two parts are shown in Figure 15 and Figure 16, respectively. As shown in Figure 14, the simulation mesh of the upper surface in Abaqus is denser than other regions. It can be seen from Figure 14c that the visualization accuracy of the R-initial intermediate mesh model is better since it is generated based on the hybrid mesh size control and has more mesh nodes than the other two mesh models. Meanwhile, the visualization accuracy of the D-initial intermediate mesh model is relatively worse as it has fewer nodes on the upper surface. However, as shown in Figure 14d, the GA-based uniform intermediate mesh model achieves very good visualization accuracy since it has more nodes on the upper surface as well. The evaluation of local results is similar to the global evaluation.



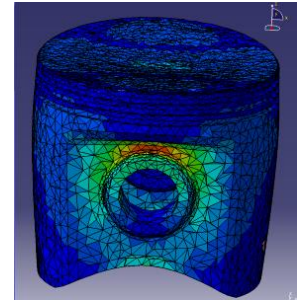
(a) Structural field simulation results in Abaqus



(b) Simulation results of D-initial mesh model in CATIA

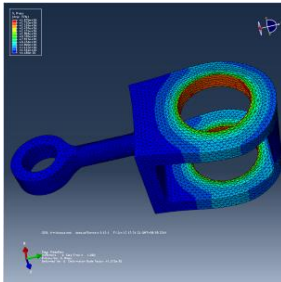


(c) Simulation results of R-initial mesh model in CATIA

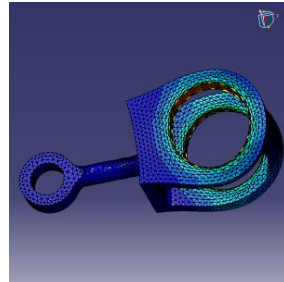


(d) Simulation results of GA-based mesh model in CATIA

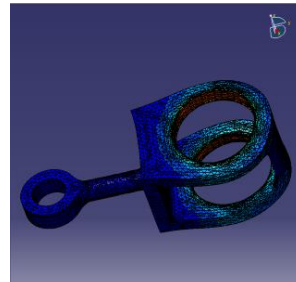
Figure 10. Structural field simulation results of the *piston* part.



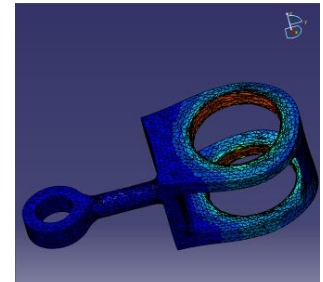
(a) Structural field simulation results in Abaqus



(b) Simulation results of D-initial mesh model in CATIA

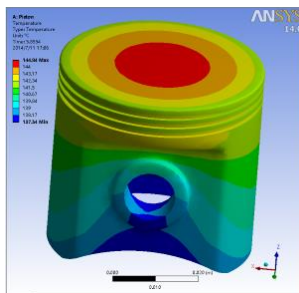


(c) Simulation results of R-initial mesh model in CATIA

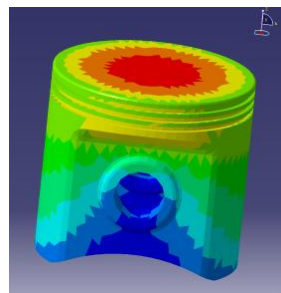


(d) Simulation results of GA-based mesh model in CATIA

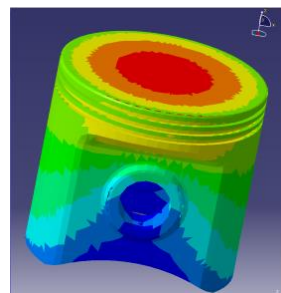
Figure 11. Structural field simulation results of the *support arm* part.



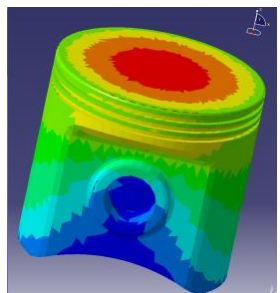
(a) Thermal field simulation results in Ansys



(b) Simulation results of D-initial mesh model in CATIA

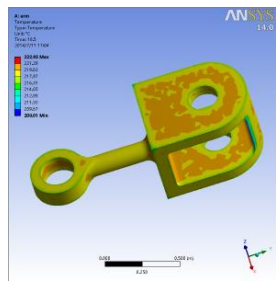


(c) Simulation results of R-initial mesh model in CATIA

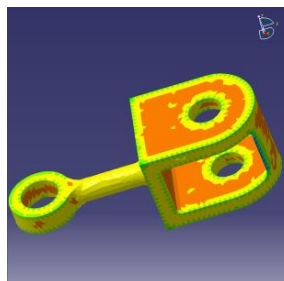


(d) Simulation results of GA-based mesh model in CATIA

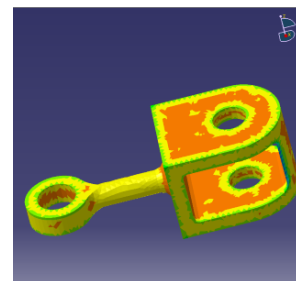
Figure 12. Thermal field simulation results of the *piston* part.



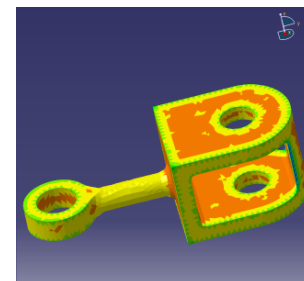
(a) Thermal field simulation results in Ansys



(b) Simulation results of D-initial mesh model in CATIA



(c) Simulation results of R-initial mesh model in CATIA



(d) Simulation results of GA-based mesh model in CATIA

Figure 13. Thermal field simulation results of the *support arm* part.

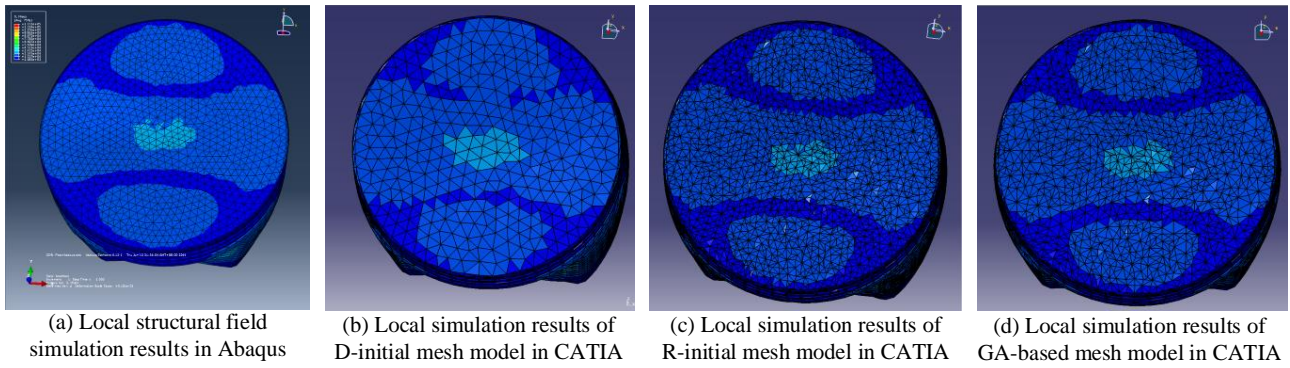
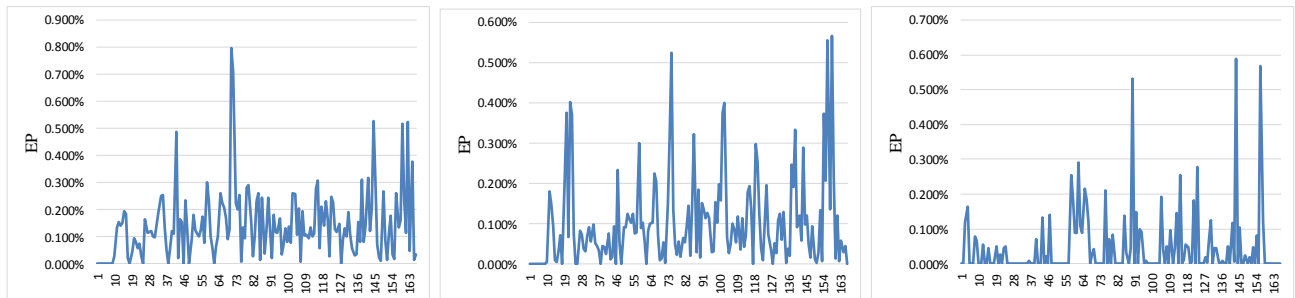


Figure 14. Simulation results of local structural field analysis for the *piston* part.

Generally, the local accuracy is higher than the global one. For example, as shown in Table 3, the node number 12420 of the *piston* part is significantly less than that of the R-initial intermediate mesh model (20445). However, for the local region, the upper surface, the SEP is increased from 79.27% to 89.90% in [0, 2.5%] as shown in Figure 16. For the thermal field analysis of the *cabinet* part, as mentioned above, the visualization accuracy of R-initial mesh model and

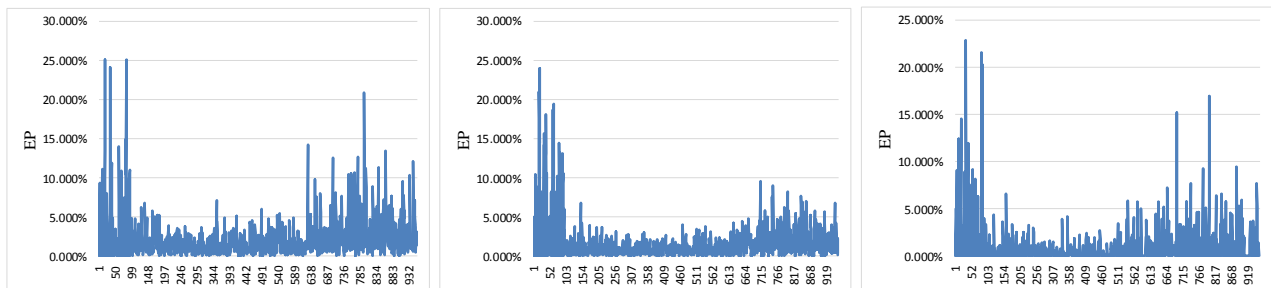
GA-based uniform intermediate mesh model are similar. In fact, the results of GA-based uniform intermediate mesh model are more accurate due to the fact that the SEP is increased from 86.23% to 94.61% in [0, 0.2%]. Based on the above analysis, the GA-based uniform intermediate mesh model can improve visualization accuracy of the local critical regions with fewer nodes.



SEP	0~0.2%	0.2%~0.4%	0.4%~0.6%	>0.6%	SEP	0~0.2%	0.2%~0.4%	0.4%~0.6%	>0.6%	SEP	0~0.2%	0.2%~0.4%	0.4%~0.6%	>0.6%
Percentage	76.05%	20.36%	2.40%	1.19%	Percentage	86.23%	10.78%	2.99%	0.00%	Percentage	94.61%	3.59%	1.80%	0.00%

(a) Local SEP of the D-initial mesh model (b) Local SEP of the R-initial mesh model (c) Local SEP of the GA-based mesh model

Figure 15. Local SEP results of different intermediate mesh models for the *cabinet* part.



SEP	0~2.5%	2.5%~5%	5%~10%	>10%	SEP	0~2.5%	2.5%~5%	5%~10%	>10%	SEP	0~2.5%	2.5%~5%	5%~10%	>10%
Percentage	67.29%	22.50%	7.71%	2.50%	Percentage	79.27%	14.27%	4.38%	2.08%	Percentage	89.90%	5.73%	3.54%	0.83%

(a) Local SEP of the D-initial mesh model (b) Local SEP of the R-initial mesh model (c) Local SEP of the GA-based mesh model

Figure 16. Local SEP results of different intermediate mesh models for the *piston* part.

8.4. Efficiency analysis

As mentioned above, the mesh node number is closely related to the errors and the time cost of mapping and interpolation. The *cabinet* part is used as the example to explain the relationship between the mesh node number and the accuracy and efficiency of visualization. By adjusting the optimization parameters in the GA-based optimization process, 22 experiments are conducted and the results are listed in Table 5 and shown in Figure 17.

It can be seen from Figure 17 that with the decrease of mesh nodes, the SEP is reduced. The time cost to obtain the simulation results is decreased as well. In other words, improvement on visualization efficiency is achieved with the cost of visualization accuracy. As shown in Table 2, the SEP of R-initial intermediate mesh model based on the hybrid mesh size control for the *cabinet* part are 79.48% in [0, 5%] and 89.26% in [0, 0.2%], respectively. Its nodes number is 38552. However, the SEP of the GA-based uniform intermediate mesh model with node number 38522 could be increased to 90.02% in [0, 5%] and 94.34% in [0, 0.2%], as shown in the 1st experiment in Table 5. As a result, the GA-based uniform intermediate mesh model could obtain a significantly better visualization result than the R-initial intermediate mesh model when they involve a similar number of nodes.

Table 5. Experiments results for the *cabinet* part with different combinations of optimization parameters α , β and γ .

No	NodeNum	Time/s			SEP		
		TAb	TAns	Average	0~5%	0~0.2%	Average
1	38522	32.592	24.068	28.330	90.02%	94.34%	92.18%
2	36031	34.174	22.289	28.232	89.15%	93.53%	91.34%
3	33793	27.683	20.911	24.297	88.08%	92.81%	90.44%
4	30846	26.532	20.427	23.480	86.31%	90.99%	88.65%
5	29093	23.705	18.205	20.955	85.48%	90.35%	87.92%
6	26671	23.007	16.891	19.949	84.37%	89.36%	86.86%
7	25253	20.717	16.440	18.579	83.38%	88.35%	85.86%
8	23366	18.762	15.125	16.944	82.32%	86.77%	84.55%
9	22327	20.397	16.530	18.464	81.55%	85.73%	83.64%
10	20864	21.173	13.222	17.198	81.11%	85.10%	83.11%
11	20098	15.634	11.462	13.548	80.49%	83.88%	82.18%
12	18990	15.326	10.835	13.081	79.82%	82.94%	81.38%
13	18491	14.387	10.774	12.581	79.67%	82.45%	81.06%
14	17639	14.089	10.554	12.322	79.10%	81.97%	80.54%
15	17326	13.641	10.529	12.085	78.77%	81.97%	80.37%
16	16702	13.987	10.215	12.101	78.77%	81.28%	80.03%
17	16492	13.772	10.136	11.954	78.37%	80.62%	79.49%
18	16008	13.332	10.207	11.770	78.02%	80.15%	79.09%
19	15885	13.365	9.804	11.585	77.95%	80.00%	78.98%
20	15490	12.822	9.721	11.272	77.65%	79.64%	78.65%
21	15438	13.007	9.649	11.328	77.63%	79.57%	78.60%
22	15141	12.634	9.454	11.044	77.38%	79.07%	78.23%

Furthermore, it can be concluded from Table 5 that a relatively small decrease in visualization accuracy could improve visualization efficiency to a large extent. For example, in the 2nd experiment, the average SEP is 91.34% while it is 84.55% in the 8th experiment. However, the time cost of the 8th experiment is reduced by 39.98% $((34.174+22.289-18.762-15.125)/(34.174+22.289))$ from the 2nd experiment. In addition, the average SEP of the 22nd experiment is 78.23%, which is 84.87% $(78.23\%/92.18\%)$ of 92.18% in the 1st experiment. However, visualization efficiency is greatly improved with a small loss in visualization accuracy. Here, the time cost is reduced by 61.02% $((32.592+24.068-12.634-9.454)/(32.592+24.068))$.

Due to the fact that the node number could be changed by adjusting the optimization parameters α and β in the GA-based optimization process, visualization accuracy and efficiency can thus be adjusted. If the simulation results need to be obtained quickly, the intermediate mesh model with fewer nodes

can be generated with a larger value of α . On the contrary, the smaller α and the larger β should be set for better visualization accuracy with a certain decrease in visualization efficiency. Therefore, the

mesh node number should be determined by the trade-off between visualization accuracy and efficiency in accordance with actual simulation requirements.

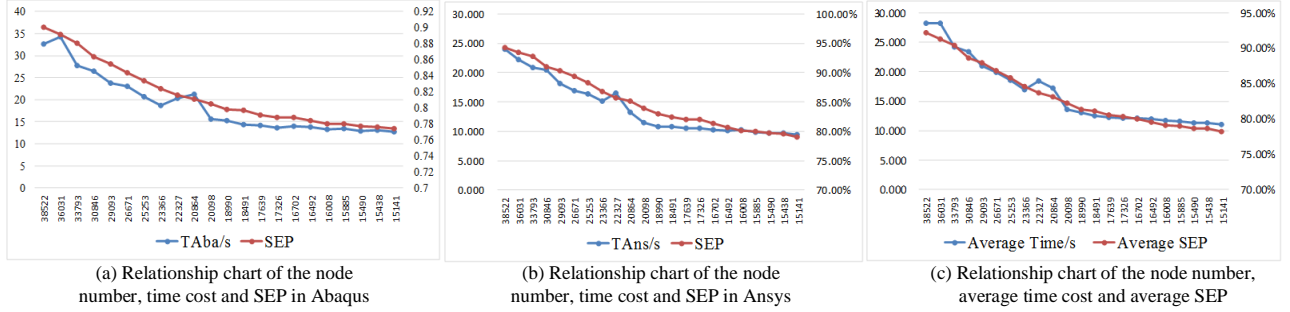


Figure 17. Time cost and SEP changing for different node numbers in the *cabinet* part simulation experiments.

9. Conclusions and Future Work

In order to support designers to conveniently and accurately view simulation results obtained using different packages for different disciplines, a uniform intermediate model is developed for efficient and high-fidelity visualization of multidisciplinary heterogeneous simulation data in CAD environments. The formal definition of the intermediate mesh optimization problem is given based on the analysis of the purpose of the uniform intermediate mesh model. In the process of solving the optimization problem, a GA-based optimization strategy is adopted and improved to generate the uniform intermediate mesh model with high accuracy and efficiency of visualization. Moreover, the hybrid mesh size field is proposed to control the generation of the R-initial mesh model to improve the algorithm's performance. The mapping and interpolation relationships between the uniform intermediate mesh model and heterogeneous simulation mesh models are established to facilitate data transfer and translation for high-fidelity visualization. Two further strategies, namely indexing and parallel computing, are devised to improve computational speed and evaluate the feasibility of the proposed method in larger real-world problems. It is shown in the evaluation work that the proposed model can eliminate the dependencies of simulation data on specific simulation tools and achieves high-fidelity and efficient visualization.

The mesh size function defined in this paper is relatively simple and rough. In our future work, more precise mesh size functions will be defined to fully reflect real mesh characteristics and on this basis strengthen mesh information description,. Meanwhile, the mapping and interpolation algorithms are relatively simple in the current implementation. Further research on more accurate mapping and interpolation algorithms are needed to improve visualization effects. Parallel computing method in this paper is only based on CPU, GPU-based parallel algorithm may also be considered in the future. In addition, more commercial CAE systems and simulation types will be considered in our future work to evaluate the method in more complicated engineering problems and dimensional reduction will be also extended.

Acknowledgements

The authors appreciate the financial supports to this research from the Natural Science Foundation of China (Grant Nos. 61173126 and 61163016), the 863 High-Technology Project of China (Grant No. 2011AA100804) and Zhejiang Provincial Natural Science Foundation(Grant No. LZ12F02001).

References

- Bazilevs, Y., Beirao da Veiga, L., Cottrell, J. A., Hughes, T. J. R., & Sangalli, G. 2006. Isogeometric analysis: approximation, stability

- and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07), 1031-1090.
- Charles, S., Bartholomew, P., & Paleczny, C. 2005. Standardization of the finite element analysis data-exchange in aeronautics concurrent engineering. *Journal of Computing and Information Science in Engineering*, 5(1), 63-66.
- Cho, S. W., Kim, S. W., Park, J. P., Yang, S. W., & Choi, Y. 2011. Engineering collaboration framework with CAE analysis data. *International Journal of Precision Engineering and Manufacturing*, 12(4), 635-641.
- Choi, D. H., Kim, T. W., & Lee, K. (2002). Multiresolutional representation of b-rep model using feature conversion. *Transactions of the Society of CAD/CAM Engineers*, 7(2), 121-130.
- Cote, F., Masson, P., Mrad, N., & Cotoni, V. 2004. Dynamic and static modelling of piezoelectric composite structures using a thermal analogy with MSC/NASTRAN. *Composite Structures*, 65(3), 471-484.
- Cottrell, J. A., Hughes, T. J., & Bazilevs, Y. 2009. *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. 2002. A fast and elitist multi-objective genetic algorithm: NSGA-II. *Evolutionary Computation*, IEEE Transactions on, 6(2), 182-197.
- Drieux, G., León, J. C., Guillaume, F., Chevassus, N., Fine, L., & Poulat, A. 2007. Interfacing product views through a mixed shape representation. Part 2: Model processing description. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 1(2), 67-83.
- Fonseca, C. M., & Fleming, P. J. 1995. An overview of evolutionary algorithms in multi-objective optimization. *Evolutionary computation*, 3(1), 1-16.
- Gujarathi, G. P., & Ma, Y. S. 2010. Generative CAD and CAE integration using common data model. In *Automation Science and Engineering (CASE)*, 2010 IEEE Conference on (pp. 586-591). IEEE.
- Gujarathi, G. P., & Ma, Y. S. 2011. Parametric CAD/CAE integration using a common data model. *Journal of Manufacturing Systems*, 30(3), 118-132.
- Hamri, O., León, J. C., Giannini, F., Falcidieno, B., Poulat, A., & Fine, L. 2008. Interfacing product views through a mixed shape representation. Part 1: Data structures and operators. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 2(2), 69-85.
- Hamri O, León J, Giannini F, et al. 2010. Software environment for CAD/CAE integration. *Advances in Engineering Software*, 41(10): 1211-1222.
- Hughes, T. J., Cottrell, J. A., & Bazilevs, Y. 2005. *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*. *Computer methods in applied mechanics and engineering*, 194(39), 4135-4195.
- Hunten, K. A. 1997. CAD/FEA integration with STEP AP209 technology and implementation. Lockheed Martin Corporation.
- Hunten, K. 2000. Enterprise analysis information integration with STEP. In *Proceedings of NASA's STEP for Aerospace Workshop*.
- Immersive SIM Engineering. 2008. URL: <http://www.immersive-sim.de>.
- ISO 10303-209. 2001. STEP - Part 209: application protocol: composite and metallic structural analysis and related design. Geneva (Switzerland).
- Jin, Y., Olhofer, M., & Sendhoff, B. 2001. Dynamic weighted aggregation for evolutionary multi-objective optimization: Why does it work and how?
- Kemmerer, S. J. (Ed.). 1999. *STEP: the grand experience*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology.
- Kumar, M., Husian, M., Upreti, N., & Gupta, D. 2010. Genetic algorithm: Review and application. *International Journal of Information Technology and Knowledge Management*, 2(2), 451-454.
- Lee, S. H., Lee, K. S., & Park, S. 2002.

- Multiresolution representation of solid models using the selective Boolean operations. In Proc. of Korean Society of Precision Engineering Conference (pp. 833-835).
- Lee, S. H. 2004. Multi-resolution modeling for feature-based solid models using the effective volumes of features. In International CAD conference and exhibition (CAD'04), Pattaya Beach, Thailand.
- Lee, S. H. 2005. A CAD–CAE integration approach using feature-based multi-resolution and multi-abstraction modelling techniques. *Computer-Aided Design*, 37(9), 941-955.
- Lee, S. H., Lee, K., & Lee, K. Y. 2005. Feature-based Multi-resolution and Multi-abstraction Non-manifold Modeling System to Provide Integrated Environment for Design and Analysis of Injection Molding Products. In Proceedings of the first Korea–China joint conference on geometric and visual computing.
- Liu, Z., Fu, Y., & Tan, J. 2010. Integration and Visual Simulation of Finite Element Data of Product Multiple Physical Fields Based on Coupling of Heterogeneous Meshes. *Journal of Mechanical Engineering*, 46(7), 114-121.
- Marler, R. T., & Arora, J. S. 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6), 369-395.
- MSC software. 2011. MSC.PATRAN release guide for version 8; URL: <http://www.mscsoftware.com/>.
- Nolan, D. C., Tierney, C. M., Armstrong, C. G., Robinson, T. T., & Makem, J. E. 2013. Automatic dimensional reduction and meshing of stiffened thin-wall structures. *Engineering with Computers*, 1-13.
- Park, B. K., & Kim, J. J. 2012. A sharable format for multidisciplinary finite element analysis data. *Computer-Aided Design*, 44(7), 626-636.
- Persson, P. O. 2006. Mesh size functions for implicit geometries and PDE-based gradient limiting. *Engineering with Computers*, 22(2), 95-109.
- Quadros, W. R., Owen, S. J., Brewer, M. L., & Shimada, K. 2004. Finite Element Mesh Sizing for Surfaces Using Skeleton. In IMR (pp. 389-400).
- Ruiz-Gironés, E., Roca, X., & Sarrate, J. 2014. Combining Size-Preserving and Smoothing Procedures for Adaptive Quadrilateral Mesh Generation. In Proceedings of the 22nd International Meshing Roundtable (pp. 19-37). Springer International Publishing.
- Schaffer, J. D., Whitley, D., & Eshelman, L. J. 1992. Combinations of genetic algorithms and neural networks: A survey of the state of the art. In *Combinations of Genetic Algorithms and Neural Networks, 1992., COGANN-92. International Workshop on* (pp. 1-37). IEEE.
- Sellgren, U. 1994. Simulation driven design-Necessity and feasibility.
- Sellgren, U. 1995. Simulation driven design - A functional view of the design process. Licentiate Thesis, Department of Machine Design, Royal Institute of Technology, Stockholm.
- Shao, Y., Liu, Y., Ye, X., & Fang, Y. 2013. CAD-Centered Integration and Efficient Visualization of Multidisciplinary Simulation Data. In *Computer-Aided Design and Computer Graphics (CAD/Graphics), 2013 International Conference on* (pp. 330-336). IEEE.
- Sivaraj, R., & Ravichandran, T. 2011. A review of selection methods in genetic algorithm. *International journal of engineering science and technology*, 3(5), 3792-3797.
- Smit, M. S., & Bronsvort, W. F. 2009. Integration of design and analysis models. *Computer-Aided Design and Applications*, 6(6), 795-808.
- Song, I. H., Yang, J., Jo, H., & Choi, S. 2009. Development of a lightweight CAE middleware for CAE data exchange. *International Journal of Computer Integrated Manufacturing*, 22(9), 823-835.
- Srinivas, M., & Patnaik, L. M. 1994. Adaptive probabilities of Crossover and mutation in genetic algorithms. *IEEE Transaction on Systems, Man, and Cybernetics*, 24(4).
- Sun, J., Garibaldi, J. M., & Hodgman, C. 2012. Parameter estimation using meta heuristics in

- systems biology: a comprehensive review. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 9(1), 185-202.
- Wei, Y., & Qiqiang, L. 2004. Survey on Particle Swarm Optimization Algorithm [J]. *Engineering Science*, 5(5), 87-94.
- Zhang, C., & Zhang, L. 2014. Model of multidisciplinary simulation integration in helicopter rotor blade design process. *International Journal of Computer Integrated Manufacturing*, 27(3), 229-241.
- Zhu, J., Blacker, T. D., & Smith, R. 2002. Background Overlay Grid Size Functions. In *IMR* (pp. 65-73).